


Article

# CORDIC-Based General Multiple Fading Generator for Wireless Channel Digital Twin

Chen Fang<sup>1</sup>, Kai Mao<sup>1</sup>, Sheng Fang<sup>1</sup>, Zikun Zhao<sup>1</sup>, Boyu Hua<sup>1</sup>, Tao Liu<sup>1</sup> and Qiuming Zhu<sup>1,2,\*</sup> 

<sup>1</sup> The Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

<sup>2</sup> National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China

\* Correspondence: zhuqiuming@nuaa.edu.cn

**Abstract:** A wireless channel digital twin is a useful tool to evaluate the performance of a communication system at the physical or link level by generating the physical channel controllably. In this paper, a stochastic general fading channel model is proposed, which considered most of the channel fading types for various communication scenarios. By using the sum-of-frequency-modulation (SoFM) method, the phase discontinuity of the generated channel fading was well addressed. On this basis, a general and flexible generation architecture for channel fading was developed on a field programmable gate array (FPGA) platform. In this architecture, improved CORDIC-based hardware circuits for the trigonometric function, exponential function, and natural logarithm were designed and implemented, which improved the real-time performance of the system and the utilization rate of the hardware resources compared with the traditional LUT and CORDIC method. For a 16-bit fixed-point data bit width single-channel emulation, the hardware resource consumption was significantly reduced from 36.56% to 15.62% for the overall system by utilizing the compact time-division (TD) structure. Moreover, the classical CORDIC method brought an extra latency of 16 system clock cycles, while the latency caused by the improved CORDIC method was decreased by 62.5%. Finally, a generation scheme of a correlated Gaussian sequence was developed to introduce a controllable arbitrary space–time correlation for the channel generator with multiple channels. The output results of the developed generator were consistent with the theoretical results, which verified the correctness of both the generation method and hardware implementation. The proposed channel fading generator can be applied for the emulation of large-scale multiple-input, multiple-output (MIMO) channels under various dynamic communication scenarios.

**Keywords:** channel digital twin; channel fading generator; CORDIC; statistical properties



**Citation:** Fang, C.; Mao, K.; Fang, S.; Zhao, Z.; Hua, B.; Liu, T.; Zhu, Q. CORDIC-Based General Multiple Fading Generator for Wireless Channel Digital Twin. *Sensors* **2023**, *23*, 2712. <https://doi.org/10.3390/s23052712>

Academic Editor: Elena Simona Lohan

Received: 18 January 2023  
Revised: 21 February 2023  
Accepted: 23 February 2023  
Published: 1 March 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Channel fading plays an important role for the realistic performance assessment of communication technologies [1]. The digital twin is becoming a key technology to optimize wireless communication algorithms and evaluate the performance of wireless communication systems [2–4]. General channel fading reproduction (or a digital twin) for arbitrary communication scenarios in the laboratory is an effective way to speed up the validation, evaluation, and optimization of new communication technologies at the physical or link level [5–7]. However, the existing channel emulator architecture based on the stochastic channel model has poor universality and is only suitable for the emulation of channel characteristics for several specific communication scenarios [8,9]. In addition, the traditional channel fading emulation methods are limited by the utilization rate of field programmable gate array (FPGA) hardware resources and the real-time performance of the algorithm when they are applied for larger-scale multiple-output, multiple-output (MIMO) channels or highly dynamic channels in B5G/6G [10,11].

There are various scenario-dependent channel fading models reported in the literature [12–16]. For example, by analyzing massive measurement data, channel fading was

described as a Nakagami distribution in the suburban scenario [12]. Due to the blocking by buildings and mountains, it was modeled in [13] as a lognormal distribution in a built-up area. For urban areas, the Weibull distribution was adopted in [14] to evaluate the network-level performance. Moreover, the compound fading types such as shadowed-Rician are suitable for the channel fading emulation of satellite and terrestrial links [15,16]. The authors in [17–19] proposed software-based simulators to generate the channel fading such as Corazza, Loo, and NLN. However, with the increase of the computational complexity, these software-based methods' results have poor real-time performance.

The traditional sum-of-sinusoids (SOS) method and its derivations were developed for real-time hardware implementation [14,20–23]. However, the phase of output fading is not continuous and smooth due to the idea of the piecewise operation in the SoS method. Li and Zhu proposed the sum-of-frequency-modulation (SoFM) method, which can ensure the continuity of the Doppler phase [20]. Several representative hardware generators based on an FPGA platform were addressed in [14,20–22,24,25]. Among them, the most popular one is based on the cosine lookup table (LUT) method [20,22]. The simple hardware structure of this method can generate various nonlinear functions. However, a huge amount of random-access memory (RAM) resource is needed for a higher emulation accuracy and a larger MIMO scale. The CORDIC algorithm is another low-cost hardware implementation method, which can realize the fixed-point calculation of various elementary functions and only consumes a small amount of RAM resource compared to the LUT method. The authors in [14] developed a Weibull and Suzuki fading generator based on the traditional CORDIC method, but it caused a long initialization latency of 16 system clock cycles due to the large number of iterations. To reduce the iteration number and time delay, the authors in [26] improved the CORDIC algorithm by using the pipeline architecture of an FPGA and reduced the system delay to 12 clock cycles. However, this method took a large update time for the channel parameters, which is not suitable for non-stationary channel generation. The authors in [21] introduced an approximated rotation factor state, and the authors in [24,25] proposed a high radix algorithm that can further reduce the iteration latency of the CORDIC algorithm. However, the hardware circuit was complex and difficult to implement. Therefore, considering the complexity of the hardware implementation and the update cycle of time-varying channel parameters, this paper proposes an improved CORDIC method to emulate arbitrary channel fading. This paper aimed to (1) fill the above research gap and develop an efficient and universal digital twin method for arbitrary channel fading, (2) propose a correlated multi-Gaussian sequence generation architecture, and (3) achieve a controllable space–time correlation for MIMO channels. Overall, the major contributions and novelties are summarized as follows:

(1) A novel FPGA-based hardware generation architecture for general channel fading in various communication scenarios was developed. The new architecture takes most of the channel fading into account, i.e., Rayleigh, Rice, Weibull, Nakagami, lognormal, Suzuki, Corazza, NLN, and Loo, and can be applied to rural, urban, suburban, and high-speed rail scenarios, especially land mobile satellite communication scenarios. Moreover, by updating the hardware parameters in real-time, the proposed architecture also supports the non-stationary channel fading under dynamic scattering scenarios with a continuous channel phase. It can be used to map the physical channel to the controllable and repeatable digital space.

(2) The CORDIC-based hardware circuits for the trigonometric functions, the exponential function, and the natural logarithm were developed, which consisted of an angle mapping module, angle recording module, and iteration module. By using a greedy CORDIC parallel pipeline structure and domain folding technique, the proposed method consumes less RAM resource and has better real-time performance than existing methods. The inherent delay caused by the improved CORDIC method of channel fading emulation is less than the parameter updating period, which is more suitable for the emulation of non-stationary channel fading. Moreover, the arbitrary space–time correlation of output channel fading can be controlled by choosing proper parameters based on the improved

hardware implementation architecture. It is beneficial for the spatial–temporal correlated channel fading generation, i.e., MIMO channel and diversity channel.

(3) The proposed generator was implemented on a Xilinx XC7K325TFFG900 FPGA chip. The hardware emulation results showed that, compared with the traditional LUT method, the hardware resource of the improved CORDIC method was reduced from 9.14% to 6.21% when a single Gaussian sequence was generated in single-channel fading. Especially, with the increasing scale of MIMO, the number of channels to be emulated increased. The reduction of the hardware resource consumption was more significant when using our proposed greedy CORDIC method. In addition, the latency was cut by 62.5% compared to the classical CORDIC method, which is vital to support non-stationary channel emulation with fast time-variant channel characteristics. The output PDF agreed well with the theoretical one, and the average deviation was below 1.63% for a 16-bit data width. Moreover, the mean absolute error was merely 0.46% for CCF, and the average deviation was about 5.62% for ACF, which verified the correctness of the designed channel emulator.

The remainder of this paper is organized as follows. In Section 2, a channel model with general fading under different scenarios is illustrated. Section 3 gives the hardware architecture and FPGA implementation. A CORDIC-based generation method is developed for generating arbitrary channel fading. Meanwhile, a correlated multiple Gaussian sequence generation scheme is also given to introduce the auto- and cross-correlations between different channel fading types. Section 4 compares the hardware resource consumption and validates the generated channel fading. Finally, some conclusions are given in Section 5.

## 2. General Fading Channel Model

Based on radio propagation theory, the received signal consists of multiple paths with different delays and angles due to reflection, refraction, and diffraction. The channel impulse response (CIR) can be modeled as

$$h(t, \tau) = \sum_{m=1}^M \{ \sqrt{P_m(t)} \cdot r_m^i(t) \cdot e^{j\phi_m(t)} \cdot \delta(\tau - \tau_m) \} + n(t) \quad (1)$$

where  $M$  is the number of valid path,  $n(t)$  is the additive channel noise, and  $P_m(t)$ ,  $r_m^i(t)$ ,  $\phi_m(t)$ , and  $\tau_m$  represent the path loss, the amplitude of channel fading such as small-scale fading and large-scale fading (or shadowing), the phase, and the delay of the  $m$ th path, respectively;  $i = \text{RL, R, Na, Log, S, W, Loo, Co, NLN}$  denote different types of channel fading.

Note that the channel fading in (1) can be modeled and generated in a deterministic or stochastic way. For the deterministic model, the fading parameters can be obtained by using the ray tracing (RT) method on specific communication scenarios. However, for some applications, we may have no detailed geometry information and the stochastic models based on measured data would be used. Table 1 gives widespread stochastic models of channel fading, which cover most of the communication scenarios. Deeply understanding and precisely reproducing these channel fading types are very useful for the exploration, design, and optimization of the communication systems.

**Table 1.** Different types of channel fading.

$i$	Channel Fading	Communication Scenario	References
RL	Rayleigh	Tall buildings	[8,10,12,13,27]
R	Rice	Suburban, country	
Na	Nakagami	Tall buildings, suburban, country	
Log	Lognormal	Buildings, mountains, other obstacles	
S	Suzuki	Urban	[9,14]
W	Weibull	Urban	
Loo	Loo	Country, suburban	[17,18]
Co	Corazza	Highway, urban, suburban, rural	
NLN	NLN	Mobile satellite	[19]

### 3. Channel Fading Generation and Implementation

#### 3.1. Overview of Hardware Generation Architecture

Due to the unique characteristics of the pipeline hardware circuit structure and parallel operation, FPGAs can reduce the complexity of hardware operation, improve the efficiency, and reduce the operation time, so they are suitable for the hardware emulation of complicated and time-variant channel fading. Considering that the channel model in (1) cannot be directly implemented in an FPGA, we firstly transferred it to the discrete model before the hardware implementation. The complex CIR can be rewritten in the discrete domain as

$$h(l, \zeta) = \sum_{m=1}^{M(l)} \{ \sqrt{P_m(l)} \cdot r_m^i(l) \cdot e^{j\phi_m(l)} \cdot \delta(\zeta - \lfloor \tau_m(l) \rfloor_{T_s}) \} + n(l) \quad (2)$$

where  $l$  and  $\zeta$  are the discrete indexes in the time and delay domains, respectively, and  $\lfloor \tau_m(l) \rfloor_{T_s}$  denotes the discrete path delay. Note that the channel fading factor  $r_m^i(t)$  in (2) is the key to reproducing different channel models. Therefore, we mainly focused on the generation of  $r_m^i(t)$  in this paper [28–30].

The Gaussian sequence is a discrete random sequence, and all elements are Gaussian variables. Let us define the  $m$ th discrete Gaussian sequence  $u_m(l) \sim N(0, \sigma_m^2)$  with the variance  $\sigma_m^2$ ; the Rayleigh fading can be expressed as

$$r_m^{\text{RL}}(l) = |u_m(l) + j \cdot \hat{u}_m(l)| \quad (3)$$

where  $l$  is the discrete sampling time and  $u_m(l)$ ,  $\hat{u}_m(l)$  represent the Gaussian sequence and its Hilbert transformation, respectively; both are independent orthogonal components of Rayleigh fading. Similarly, Rice, Weibull, and Nakagami fading can be expressed as

$$r_m^{\text{R}}(l) = |\rho_m \cdot \exp(j(2\pi f_{m,\rho} l + \theta_{m,\rho})) + u_m(l) + j \cdot \hat{u}_m(l)| \quad (4)$$

$$r_m^{\text{W}}(l) = \exp(w_m / 2 \cdot \ln(r_m^{\text{RL}}(l))) \quad (5)$$

$$r_m^{\text{Na}}(l) = \sqrt{u_{m,1}^2(l) + u_{m,2}^2(l) + \dots + u_{m,2q}^2(l)} \quad (6)$$

where  $\rho_m$ ,  $f_{m,\rho}$ , and  $\theta_{m,\rho}$  are the amplitude, Doppler frequency, and phase of the line-of-sight path of Rice fading, respectively,  $w_m$  is the shape factor of Weibull fading, and  $2q$  denotes the number of independent Gaussian sequences of Nakagami fading.

Furthermore, the lognormal fading can be expressed as

$$r_m^{\text{Log}}(l) = \exp(\sigma_m^{\text{Log}} u_m(l) + \mu_m^{\text{Log}}) \quad (7)$$

where  $\sigma_m^{\text{Log}}$  and  $\mu_m^{\text{Log}}$  are the standard deviation and mean value of lognormal fading, respectively. Finally, Suzuki, Corazza, NLN, and Loo fading can be obtained by the combination of (2)–(6) as

$$r_m^{\text{S}}(l) = r_m^{\text{RL}}(l) r_m^{\text{Log}}(l) \quad (8)$$

$$r_m^{\text{Co}}(l) = r_m^{\text{R}}(l) r_m^{\text{Log}}(l) \quad (9)$$

$$r_m^{\text{NLN}}(l) = r_m^{\text{Na}}(l) r_m^{\text{Log}}(l) \quad (10)$$

$$r_m^{\text{Loo}}(l) = |r_m^{\text{Log}}(l) \rho_m \cdot \exp(j(2\pi f_{m,\rho} l + \theta_{m,\rho})) + u_m(l) + j \cdot \hat{u}_m(l)| \quad (11)$$

The Hardware structure of general channel fading generation is shown in Figure 1. Based on (2)–(10), the  $\phi_{m,n}$  of the  $n$ th ray within the  $m$ th valid path, the amplitude  $\rho_m$ , frequency  $f_{m,\rho}$ , initial phase  $\theta_{m,\rho}$  of the line-of-sight path, and other controlling factors  $\sigma_m^{\text{Log}}$ ,

$\mu_m^{\text{Log}}$ , and  $w_m$  can be configured by the user via a periphery interface. Note that the system only consists of several multipliers, accumulators, adders, and key modules for specific functions, i.e., trigonometric function, exponential function, natural logarithm function, and correlated multiple Gaussian sequences generation, which is used to implement the correlation matrix operation (CMO). The final output is controlled by a multiplexer according to the user-defined fading type.

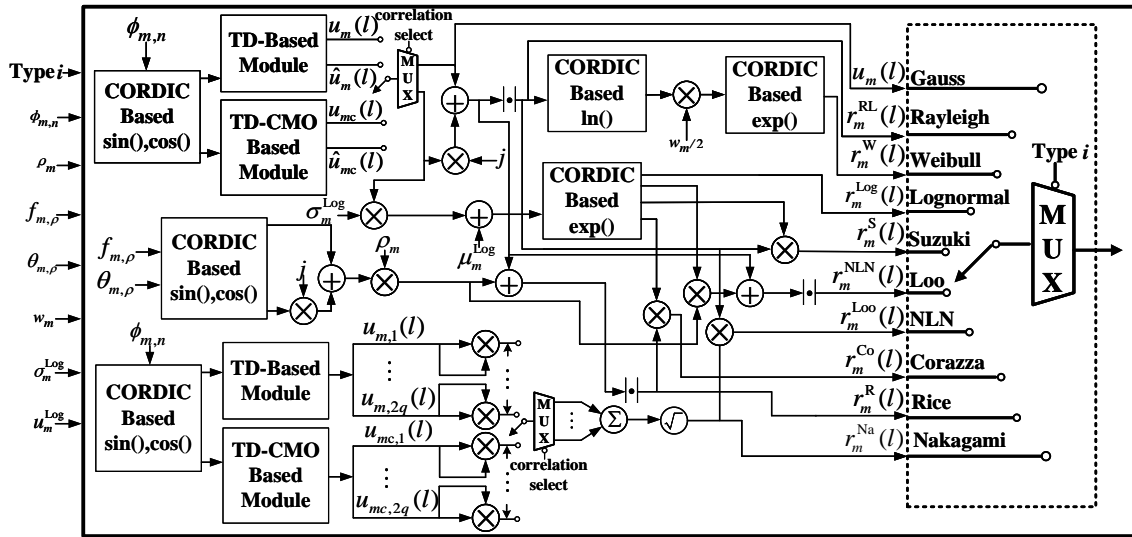


Figure 1. Hardware structure of general channel fading generation.

### 3.2. CORDIC-Based Trigonometric Operation

The Gaussian sequence is the key to the proposed hardware architecture, so it is essential to develop an efficient hardware algorithm to generate multiple Gaussian variables. Based on the central limit theorem, a Gaussian random sequence can be modeled as the superposition of several cosine waves with random initial phases and different frequencies. Different from the traditional pseudo-random-bit-based generator, it can control the spectrum shape by choosing proper phase and frequency parameters [23]. Thus, an efficient trigonometric operation for cosine calculation is needed. So far, the LUT is the most-widespread method due to its good real-time performance, but it consumes much memory resource. For the B5G/6G communication system evaluation, the channel emulator needs to emulate non-stationary channel scenarios, where the channel parameters change dynamically in real-time [7]. The CORDIC module is initialized after the parameters are updated, which would cause latency for the valid channel fading output every time. Therefore, the system latency should be reduced as much as possible in emulating fast time-variant non-stationary channels. The traditional CORDIC algorithm consumes less hardware resource, but has the shortcoming of high latency with a 16-system-clock-cycle delay.

The basic idea of CORDIC is to decompose the trigonometric operation into a set of linear combinations of micro-rotation angles, as shown in Figure 2. A two-dimensional vector  $(x_0, y_0)$  is rotated by an angle  $\theta$  to calculate the rotated vector  $(x_1, y_1)$  under the two rotation models. This transformation can be expressed, respectively, as [31]

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cosh \theta & -\sinh \theta \\ \sinh \theta & \cosh \theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (13)$$

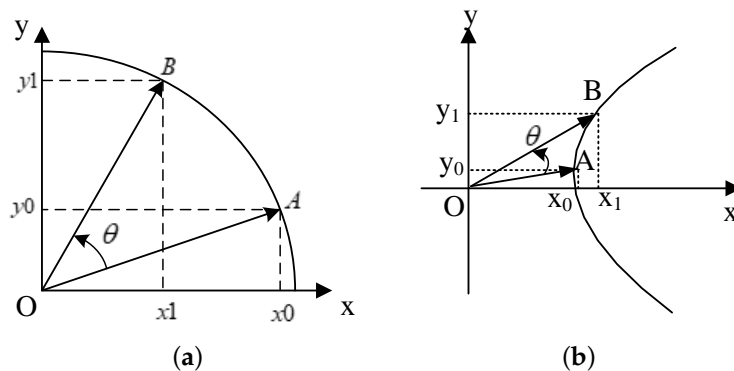


Figure 2. (a) Circular and (b) hyperbolic rotation model.

Using the traditional CORDIC method to generate the trigonometric operation requires 16 iterative operations, so it will bring 16 clock cycles of latency [24], which is unacceptable for the real-time performance of the system. To tackle the issue of the high latency due to multiple iterations, a greedy CORDIC-based trigonometric operation is developed in this section. The target angle  $\theta$  is encoded as a linear combination of the micro-rotation angle set denoted by  $\{0 \leq k \leq L - 1 | \theta(k)\}$ . The proposed method contains three main parts, i.e., angle mapping module, angle recording module, and iteration module.

The angle mapping module aims to map the input target phase  $\phi_{m,n} \in (0, 2\pi]$  to a smaller angle range. It yields a significant advantage of computational complexity when narrowing the angle range. In this paper, we divided the input angle into 16 equal domains with the uniform span of  $\pi/8$ . Let us take the first and second quadrants as an example: angle mapping can be realized through binary coding based on the domain folding, as shown in Figure 3.

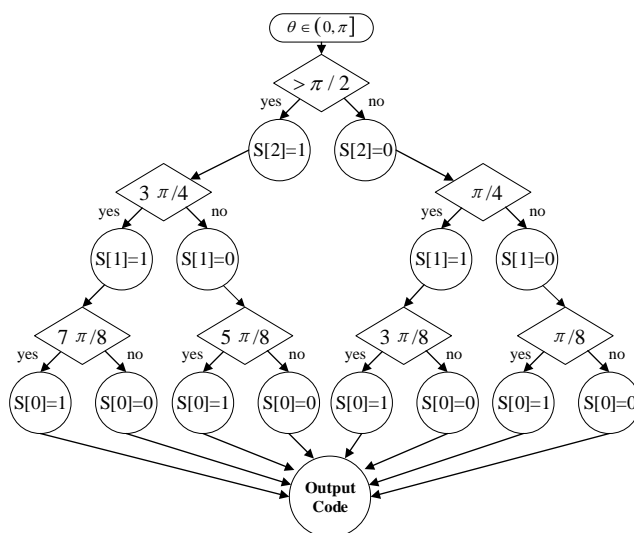


Figure 3. Angle mapping via binary coding.

The original input angle phase can be mapped to  $\phi \in [0, \pi/8]$  by the angle mapping, then we can reduce the iteration number by reducing the redundant computation of unnecessary micro-rotation angles in the iteration process. The greedy-based angle recording module is shown in Figure 4. Based on the set of micro-rotation angles  $\{0 \leq k \leq L - 1 | \theta(k)\}$ , we used  $L$  subtractors and one multiplexer to express the mapped angle  $\phi$  with several micro-rotation angles and  $L$ -bit codes  $c_1, c_2$ , where  $c_1, c_2$  are used to compute the scaling factor, the direction of the sign function, and the shift value in the iteration unit. This process adopts a parallel pipeline structure to ensure  $c_1, c_2$  are generated within one clock

cycle. Since the magnitude of the sign function is 1 and the sequence  $\vartheta(i)$  is decreasing, the convergence of this process can be proven.

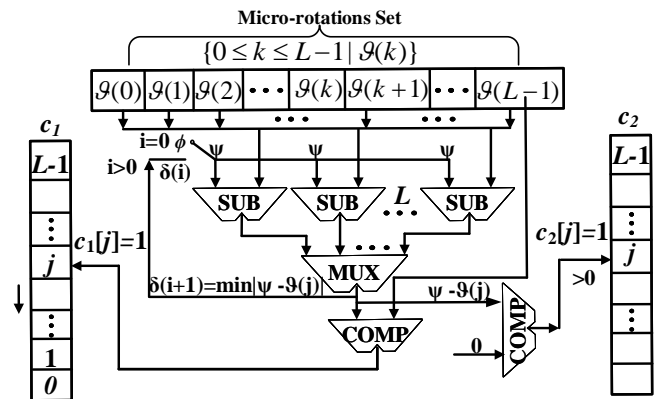


Figure 4. Greedy-based angle recording module.

By factoring  $\cos \theta$  and the approximation equation  $\tan(\vartheta(k)) \approx 2^{-k}$ , the calculation can be simplified by several shifts and add operations. Then, we can obtain the direction of sign function  $d_i$  and shift value  $b_i$  by determining the value of each bit within  $c_1, c_2$  in the  $i$ th iteration process. Figure 5 shows one stage of the iterative structure, and it can be described as

$$\begin{aligned} x_{i+1} &= (x_i - d_i \cdot y_i \cdot 2^{-b_i}) \\ y_{i+1} &= (y_i + d_i \cdot x_i \cdot 2^{-b_i}) \end{aligned} \tag{14}$$

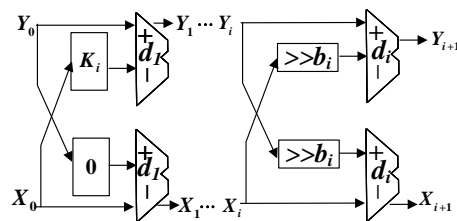


Figure 5. CORDIC-based circuit of the trigonometric function.

To calculate the trigonometric function, the values of  $x_0$  and  $y_0$  were initialized, respectively, to  $K$  and  $0$ . To maintain the accuracy and avoid overflow, the scaling factor  $K$  can be further expressed as

$$K = \prod_{c_1(j) \neq 0} \cos \vartheta(j) = \prod_{c_1(j) \neq 0} 1 / \sqrt{1 + 2^{-2j}}. \tag{15}$$

By combining (11) and (13), it can be proven that the final output after several iterations can be expressed as

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \frac{1}{x_0} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} \cos \phi \\ \sin \phi \end{bmatrix} \tag{16}$$

After several iterations, the output values of  $x_n$  and  $y_n$  are obtained to be  $\cos \phi$  and  $\sin \phi$ , respectively. By combing the mapping relationship in the angle mapping module, we can recover the trigonometric function value of  $\phi_{m,n}$ .

### 3.3. CORDIC-Based Exponential and Natural Logarithm Operation

In this paper, the CORDIC-based exponential operation was implemented in the rotation mode of (12). The input phase for computing the exponential value was obtained by decomposing the rotation through a set of micro-rotation angles denoted by

$\{1 \leq k \leq L - 1 | \vartheta(k) = \tanh^{-1}(2^{-k})\}$ . Similarly, by factoring  $\cosh \theta$  and  $\tanh(\vartheta(k)) \approx 2^{-k}$ , the iteration process can be expressed as

$$\begin{aligned} \begin{bmatrix} x_n \\ y_n \end{bmatrix} &= \cosh \theta \begin{bmatrix} 1 & \tanh \theta \\ \tanh \theta & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \\ &= \prod_{i=1}^n \begin{bmatrix} 1 & d_i 2^{-i} \\ d_i 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} K_e \end{aligned} \tag{17}$$

where  $d_i$  denotes the direction of the sign function with value  $\pm 1$ , and  $K_e$  represents the scaling factor and can be further expressed as

$$\begin{aligned} K_e &= \prod_{i=1}^n \cosh \vartheta(i) = \prod_{i=1}^n \sqrt{1 / (1 - \tanh^2 \vartheta(i))} \\ &= \prod_{i=1}^n \sqrt{1 / (1 - 2^{-2i})} \end{aligned} \tag{18}$$

For the hyperbolic CORDIC, it should be noted that, when the iterative sequence number equals  $k_n (k_n = 3k_{n-1} + 1, k_1 = 4, n \in \mathbf{Z}^+)$ , the corresponding stage of iterations should be executed again to ensure the algorithm converges [31]. The iterative recursive equation can be derived as

$$\begin{aligned} x_{n+1} &= x_n + d_n \cdot y_n \cdot 2^{-n} \\ y_{n+1} &= y_n + d_n \cdot x_n \cdot 2^{-n} \\ z_{n+1} &= z_n - d_n \cdot \vartheta(n) \end{aligned} \tag{19}$$

To calculate the exponential function, the values of  $x_0$ ,  $y_0$ , and  $z_0$  are initialized, respectively. Let us take the lognormal fading as an example: we set  $x_0 = 1, y_0 = 0$ , and  $z_0 = \sigma_m^{\text{Log}} u_m(l) + \mu_m^{\text{Log}}$ , respectively. After several iterations, when  $z_n$  equals 0, the output values of  $x_n$  and  $y_n$  can be obtained as  $\cosh(\sigma_m^{\text{Log}} u_m(t) + \mu_m^{\text{Log}})$  and  $\sinh(\sigma_m^{\text{Log}} u_m(t) + \mu_m^{\text{Log}})$ . Then, the output of  $r_m^{\text{Lo}}(t)$  can be obtained by the sum of  $x_n$  and  $y_n$ .

The natural logarithm operation has a similar architecture as the exponential operation, except the operation of the mode. In the vectoring mode, the direction of  $d_n$  is determined by the sign function of  $y_n$ . Based on the equation of  $\ln(\theta) = 2 \tanh^{-1}(\theta - 1 / \theta + 1)$ , the natural logarithm function can be calculated by setting different initial values, i.e.,  $x_0 = \theta + 1, y_0 = \theta - 1$ , and  $z_0 = 0$ . Taking Weibull fading as an example, the initial values were set as  $x_0 = |r_m^{\text{RL}}(t)| + 1, y_0 = |r_m^{\text{RL}}(t)| - 1$ , and  $z_0 = 0$ , respectively. After several iterations, when  $y_n$  equals 0, the output is  $z_n = \tanh^{-1}((|r_m^{\text{RL}}(t)| - 1) / (|r_m^{\text{RL}}(t)| + 1))$  and  $r_m^{\text{W}}(l)$  can be obtained by the definition in Table 1.

Figure 6 gives one stage of the CORDIC-based structure for the exponential and natural logarithm operations. The sign function is determined by the remaining angle  $Z_j$  in each iteration. The shift value is also determined by the iterative sequence number in each iteration. Note that the architecture uses a parallel pipeline structure to ensure that all the variables of each stage can be generated within a clock cycle.



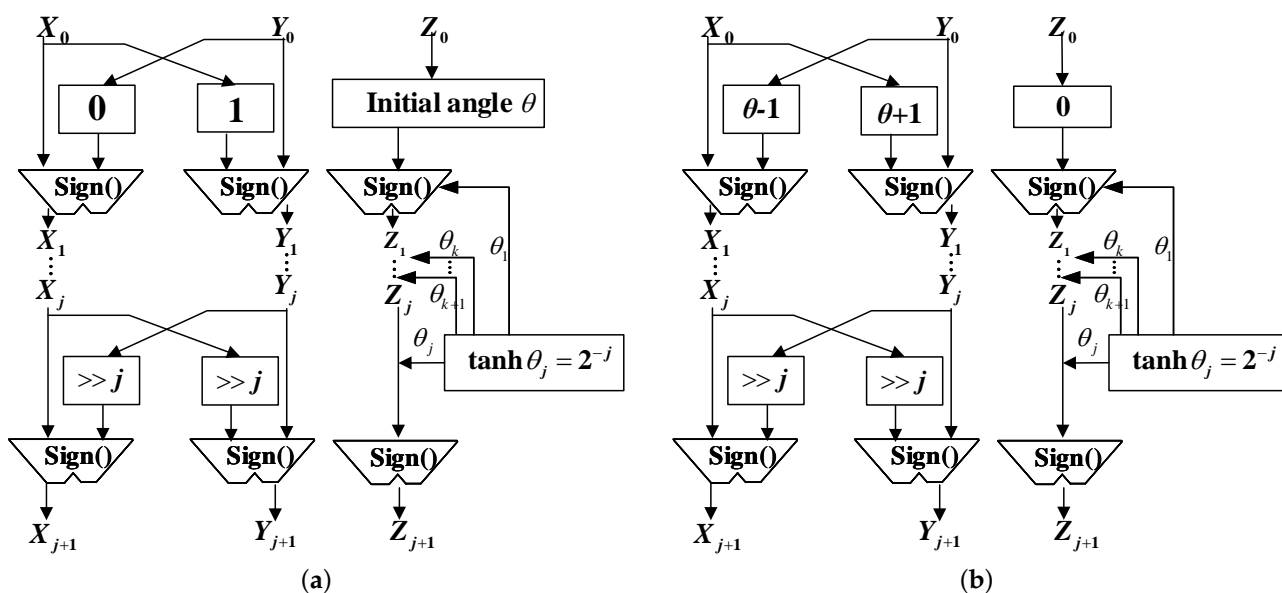


Figure 6. CORDIC-based circuits of (a) exponential and (b) natural logarithm operations.

### 3.4. Correlated Multiple Gaussian Sequence Generation

The temporal autocorrelation and spatial cross-correlation of multiple channels are very important, especially for the antenna arrangement, placement, and evaluation of MIMO systems [32]. To control the correlation of different channels or paths, we developed a correlated multiple Gaussian sequence generation scheme, as shown in Figure 7. It consists of three parts, i.e., the Doppler frequency computation, a time-division (TD)-based Gaussian unit, and the correlation matrix operation.

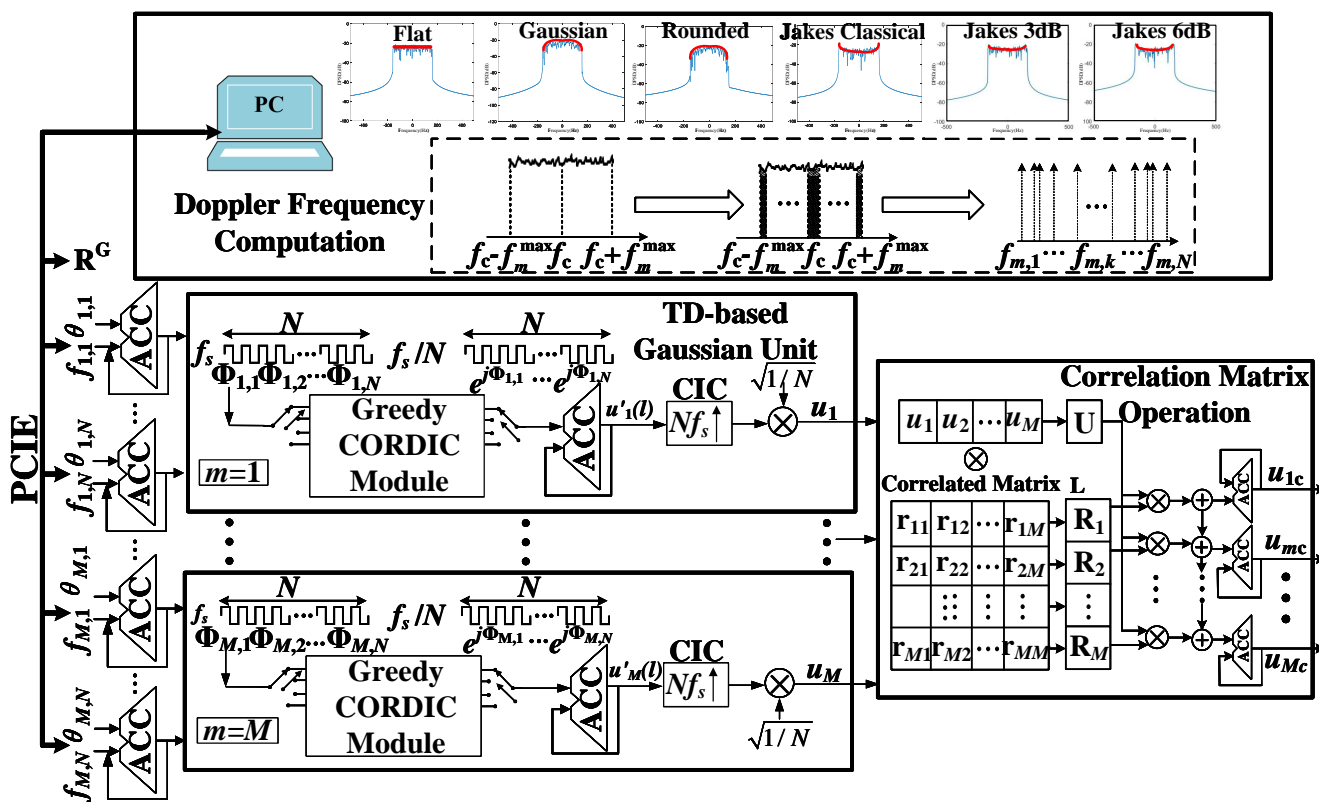


Figure 7. Overview of correlated multiple Gaussian sequence generation.

The SoS or sum-of-cisoids (SoC) method is widely adopted to generate channel fading under stationary scenarios. In order to take the non-stationary aspects into account, the dynamic update of the channel parameters is necessary for the non-stationary channel emulation under highly mobile scenarios such as high-speed trains and UAV communications. Different from the method of a non-stationary channel, it is emulated by a traditional segmented stationary channel, which updates the channel parameters at each stationary channel stage [33,34]. However, the continuity of the channel phase will not be guaranteed by this method. In order to precisely reproduce the non-stationary aspects, it is important to confirm the continuity of the output channel fading between adjacent channel states. Therefore, the channel parameters are updated in the discrete domain by the SOFM method as [20,35]

$$u_m(l) = \sum_{n=1}^N \frac{1}{\sqrt{N}} e^{\left(2\pi \sum_{k=0}^l T_s f_{m,n}[k] + \theta_{m,n}\right)} \quad (20)$$

where  $N$  is the number of branches and  $f_{m,n}$  and  $\theta_{m,n}$  are the discrete Doppler frequency and initial phase uniformly distributed over  $(0, 2\pi]$ , respectively.

Note that the discrete Doppler frequency in (19) controls the autocorrelation of the generated sequence. and it is determined by user-defined information, i.e., the DPSD power spectral density  $S_{\mu_m \mu_m}(f)$ , the carrier frequency  $f_c$ , and the maximum Doppler frequency  $f_m^{\max}$ . We firstly sampled the continuous input DPSD with a short time interval. The Doppler frequencies  $f_1, f_2 \cdots f_N$  were calculated by the MEA method, which makes the area under the DPSD curve within the range  $[-f_m^{\max}, f_m^{\max}]$  equal to  $\sigma_{m,u}^2/N$ .

$$\int_{-f_m^{\max}}^{f_m^{\max}} S_{\mu_m \mu_m}(f) df = \frac{\sigma_{m,u}^2}{N}, n = 1, 2, \dots, N \quad (21)$$

The Gaussian generation unit based on the greedy-based CORDIC module is shown in Figure 7. In order to reduce the hardware resource consumption due to massive branches, it was implemented using a TD scheme. The input phase of  $N$  rays is obtained by the Doppler frequency and initial phase, which are loaded via a high-speed transmission interface such as the PCIE bus. For each valid path, all  $N$  branches are calculated by the greedy CORDIC module and summed up by an ACC serially. The data rate of ACC output  $u'_m(l)$  is reduced to one over  $N$  of the initial rate. To match the data rate,  $u_m(l)$  is interpolated  $N$ -times with a cascade integrator comb (CIC) filter. Similarly, this TD scheme can be applied to  $M$  valid paths through a multiplexer.

On the other hand, a correlation matrix  $\mathbf{R}^G$  is introduced to control the cross-correlation between different Gaussian sequences, which can also improve the performance of ergodicity and orthogonality. The conventional correlated matrix  $\mathbf{R}^G$  is decomposed into a low triangular matrix  $L$  by the Cholesky decomposition. We found that it only works for the positive correlation matrix. To overcome this shortcoming, the eigenvalue decomposition method was adopted in this paper as

$$\begin{aligned} \mathbf{R}^G &= \mathbf{V} \bar{\Lambda} \bar{\Lambda}^H \mathbf{V}^H, \Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_M) \\ \bar{\Lambda} &= \sqrt{\text{diag}(\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_M)}, \hat{\lambda}_j = \begin{cases} \lambda_j & \lambda_j \geq 0 \\ 0 & \lambda_j < 0 \end{cases} \end{aligned} \quad (22)$$

where  $\lambda_j$  is the eigenvalue of matrix  $\mathbf{R}^G$ . Then, the decomposed matrix  $\mathbf{L} = \mathbf{V} \bar{\Lambda}$  is multiplied by the spatial independent Gaussian sequences. The cross-correlated Gaussian sequences can be obtained by  $\mathbf{U} = [u_{1c} u_{2c} \cdots u_{Mc}]$  and  $L$ .

To illustrate the above generation method, we took four temporally and spatially correlated Gaussian sequences denoted by  $h_{11}, h_{12}, h_{21}, h_{22}$  as an example. The DPSDs of the four sequences were assumed to be the same as the classical Jakes shape. The desired cross-correlation matrix  $\mathbf{R}^G$  is given as

$$\begin{bmatrix} 1 & 0.3 & 0.91 & 0.273 \\ 0.3 & 1 & 0.273 & 0.91 \\ 0.91 & 0.273 & 1 & 0.3 \\ 0.273 & 0.91 & 0.3 & 1 \end{bmatrix} \quad (23)$$

After implementing the method of Figure 7 on an FPGA chip, we exported the output fading data using the Xilinx Chipscope tool and analyzed them using the MATLAB software. The sampling rate and period were 100 MHz and 20 ms, respectively, e.g.,  $2 \times 10^6$  samples. Figure 8 depicts the amplitude of the four Gaussian sequences and the corresponding ACF. As we can see, the generated ACFs provided a quite good approximation to the theoretical ones, which can be expressed by the zero-order Bessel functions [36]. The average deviation of the ACFs between the theoretical and generated ones was about 1.26%, which can meet the wireless channel digital twin requirement.

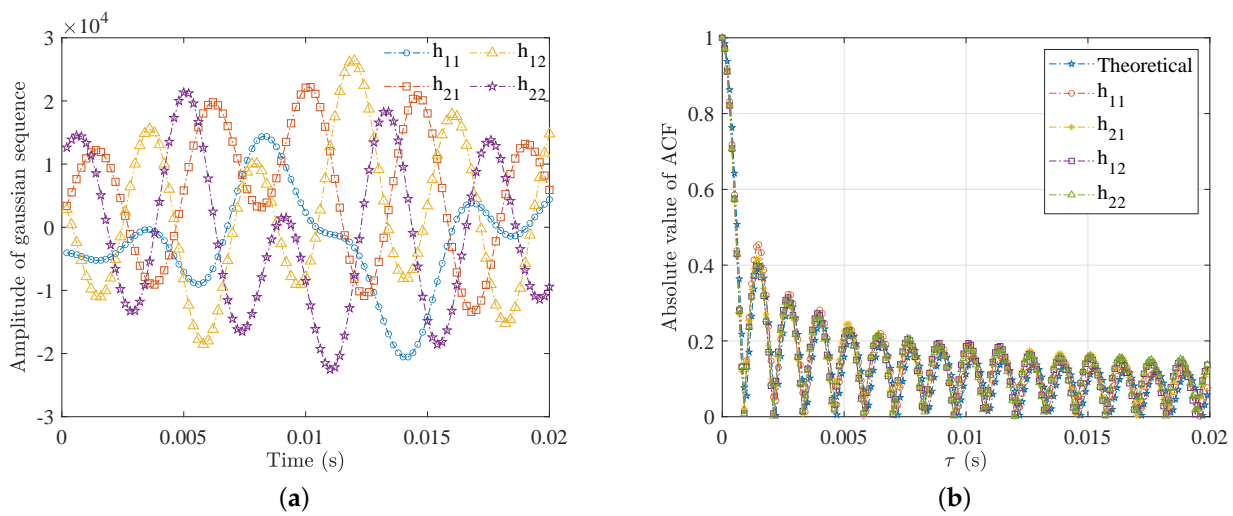


Figure 8. (a) Amplitude and (b) ACFs of the four Gaussian sequences.

Furthermore, the average CCF of the generated channels can be calculated as

$$\begin{bmatrix} 1.0000 & 0.2975 & 0.9101 & 0.2706 \\ 0.2975 & 1.0000 & 0.2676 & 0.9097 \\ 0.9101 & 0.2676 & 1.0000 & 0.2959 \\ 0.2706 & 0.9097 & 0.2959 & 1.0000 \end{bmatrix} \quad (24)$$

which also exhibited good approximation to the desired one as (13). The mean error was merely 0.46%; the maximum error was about 1.13%; the minimum error was less than 0.01%.

#### 4. Measurement Results and Validation

To validate and evaluate the proposed CORDIC-based fading generation method, we implemented the proposed arbitrary channel fading generator on a Xilinx FPGA (XC7K325TFF G900) hardware platform, where the channel consisted all the fading types in Figure 1. The system clock was 100 MHz, and the output fading data width was 16-bit. Firstly, the theoretical simulation and hardware measurement results of the DPSDF of different types of channel fading were carried out as shown in Figures 9 and 10. The small-scale channel fading of Rayleigh, Rice, Weibull, and Nakagami, the large-scale channel fading, and their derived fading groups, lognormal, Suzuki, Corazza, NLN, and Loo, were observed. The output of the channel emulator was divided into two groups to set four and five diameters, respectively, among which the Doppler spread of each multipath was set at 60 KHz. The measured spectrum shapes were consistent with those of the theoretical ones, which verified the correctness of the developed channel fading generator [37].

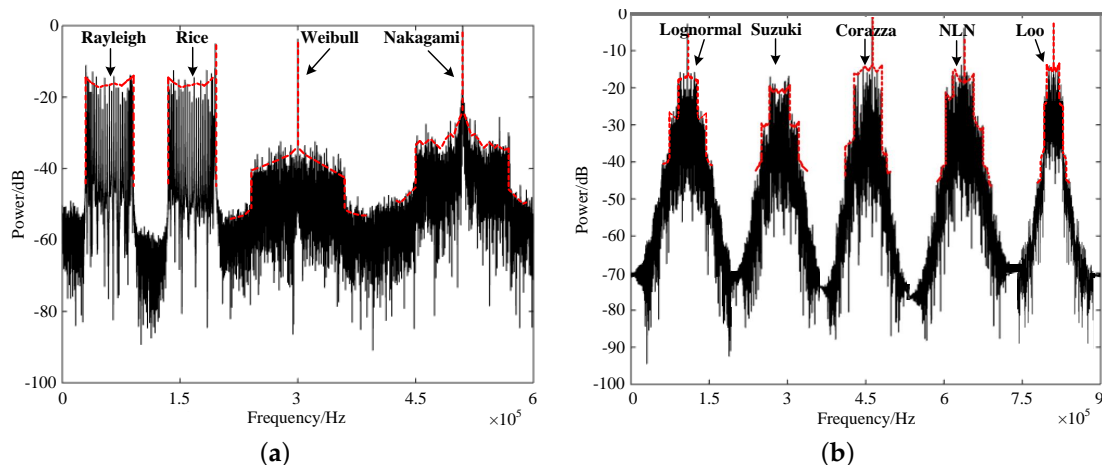


Figure 9. DPSD simulation results of (a) small-scale fading and (b) large-scale fading.

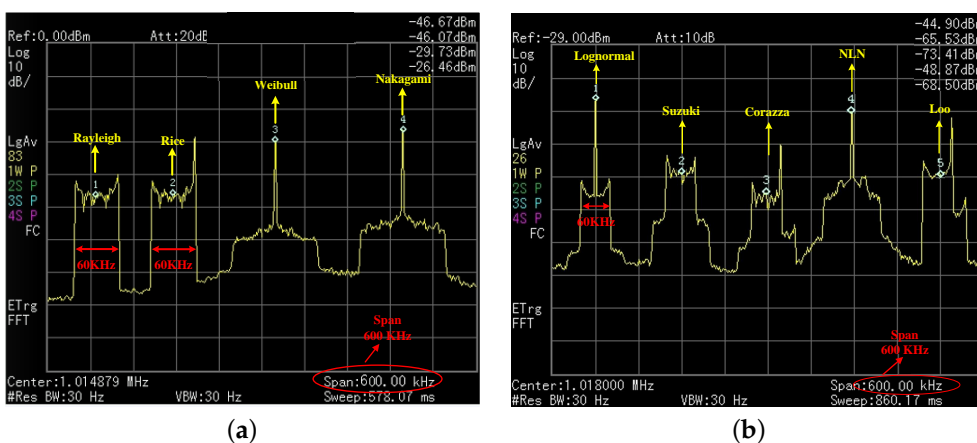


Figure 10. DPSD measurement results of (a) small-scale fading and (b) large-scale fading.

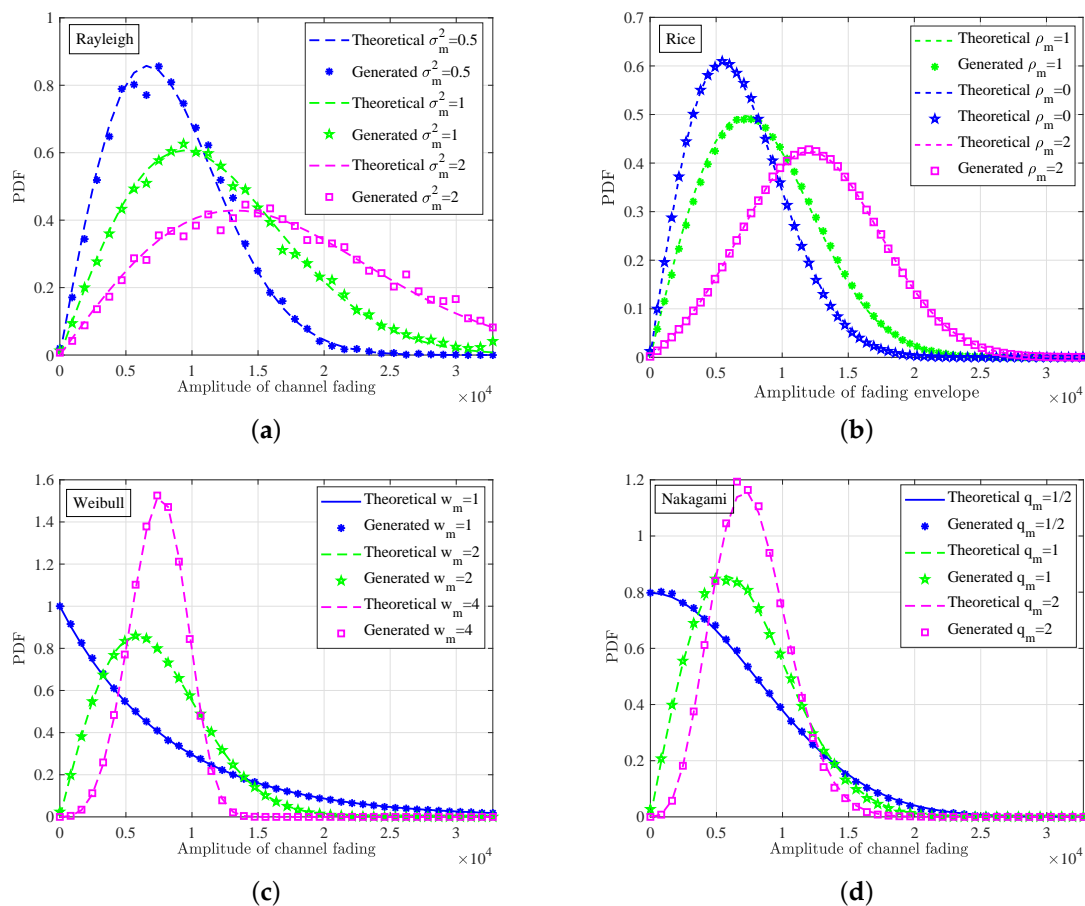
Further, Table 2 summarizes the hardware consumption in terms of the slice LUT, register, block RAM, and DSP. For comparison purposes, the hardware iteration latency is also given in Table 2 compared with the previous methods in [14,20,25,35]. As we can see, the CORDIC method had an obvious advantage with respect to the RAM consumption over the LUT methods. Note that the LUT methods needed 66 additional block RAMs (36 Kb each) to implement the trigonometric, exponential, and natural logarithm operations. By the contrast, the CORDIC-based methods needed no RAM for these operations and only consumed 18.5 block RAMs for the path delay emulation and parameter storage, i.e., the frequency, phase of each ray, and line-of sight path.

Table 2. Hardware resource consumption and latency comparison.

	[20]	Proposed	[14]	[35]	[25]
Method	LUT	CORDIC	CORDIC	CORDIC	CORDIC
Fading number	1	10	2	1	—
System clock	100 MHz	100 MHz	100 MHz	256 MHz	—
Slice LUTs	14,672	17,685	17,920	—	—
Registers	13,195	17,034	19,940	—	—
Block RAMs	84.5	18.5	18.5	—	—
DSPs	59	66	50	—	—
Utilization	9.14%	6.21%	5.95%	—	—
Average iterations	—	5.5	16	5.5	8
Critical latency	—	6Tclk	16Tclk	6Tclk	8Tclk

The iteration latency can be estimated by  $k \times l \times (T_{\text{Add}} + T_{\text{reg}} + T_{\text{shift}}) \approx kT_{\text{clk}}$ , where  $k$  is the number of iterations,  $T_{\text{Add}}$ ,  $T_{\text{reg}}$ , and  $T_{\text{shift}}$  denote the delay for the 1-bit carry look-ahead adder, register, and shift of the control signal in the pipeline structure, respectively, and  $T_{\text{clk}}$  is the basic delay of the iteration unit. In other words, the latency approximately depends on the average iteration number. As we can see, our upgraded CORDIC method reduced the iteration number to six [14,25].

According to the fluctuation rate of channel fading in the time domain, it can be divided into two kinds, i.e., small-scale fading and large-scale fading. In the following, we validated the output PDFs of the small-scale fading, i.e., Rayleigh, Rice, Nakagami, and Weibull. The sampling rate was 100 MHz; the maximum Doppler frequency  $f_m^{\text{max}}$  was 50 Hz with the DPSD of the Jakes shape; the emulation time was 1 ms. Figure 11 shows the PDFs of  $2 \times 10^6$  generated small-scale fading samples for five different conditions, i.e., Gauss, Rayleigh, Rice, Nakagami, and Weibull. Taking the Rayleigh fading of  $\sigma_m^2 = 1$  as an example, the theoretical mean and variance can be calculated by  $\sigma_m \sqrt{\pi/2}$  and  $\sigma_m^2(2 - \pi/2)$  with the fixed-point width of 16-bit, which were 41,067 and 4,608,258,070, respectively. By stating the fading data exported from the hardware platform, the measured fixed-point mean and variance were 40,887 and 4,522,732,683, respectively, and the relative error was about 0.44% and 1.86%, respectively. Overall, the average absolute deviation of the PDFs between the theoretical and generated results in Figure 11 was approximately 1.62%.



**Figure 11.** PDFs of (a) Rayleigh, (b) Rice, (c) Weibull, and (d) Nakagami fading envelope.

The large-scale fading with a lognormal distribution usually occurs with different small-scale fading, which is also called composite fading [38,39], i.e., Suzuki, Loo, Corazza, and NLN. Composite fading is useful for the digital twins of terrestrial and satellite mobile channels. Figure 12 gives the PDFs of four typical composite fading types, i.e., lognormal, Suzuki, Loo, Corazza, and NLN. As we can see, the PDFs obtained from the generated

$2 \times 10^6$  channel fading samples were close to the theoretical ones. Moreover, the minimum and maximum deviations between the theoretical and generated ones were 0.0072% and 4.36%, respectively. The average deviation was about 1.63%, which proved the effectiveness and correctness of the proposed method, as well as the hardware implementation. Taking the Suzuki fading as an example, the theoretical mean and variance can be calculated by  $\sigma_m \sqrt{\pi/2} e^{\sigma_m^2/2 + \mu_m^{\text{Log}}}$  and  $\sigma_m^2 e^{2\mu_m^{\text{Log}} + \sigma_m^{\text{Log}}} (2e^{\sigma_m^{\text{Log}}} - \pi/2)$  with the fixed-point width of 16 bits as 48,515 and 1,828,733,997, respectively. The measured variance and mean of fixed-point hardware data were 49,213 and 1,871,364,131, and the relative error was about 2.44% and 1.44%, respectively.

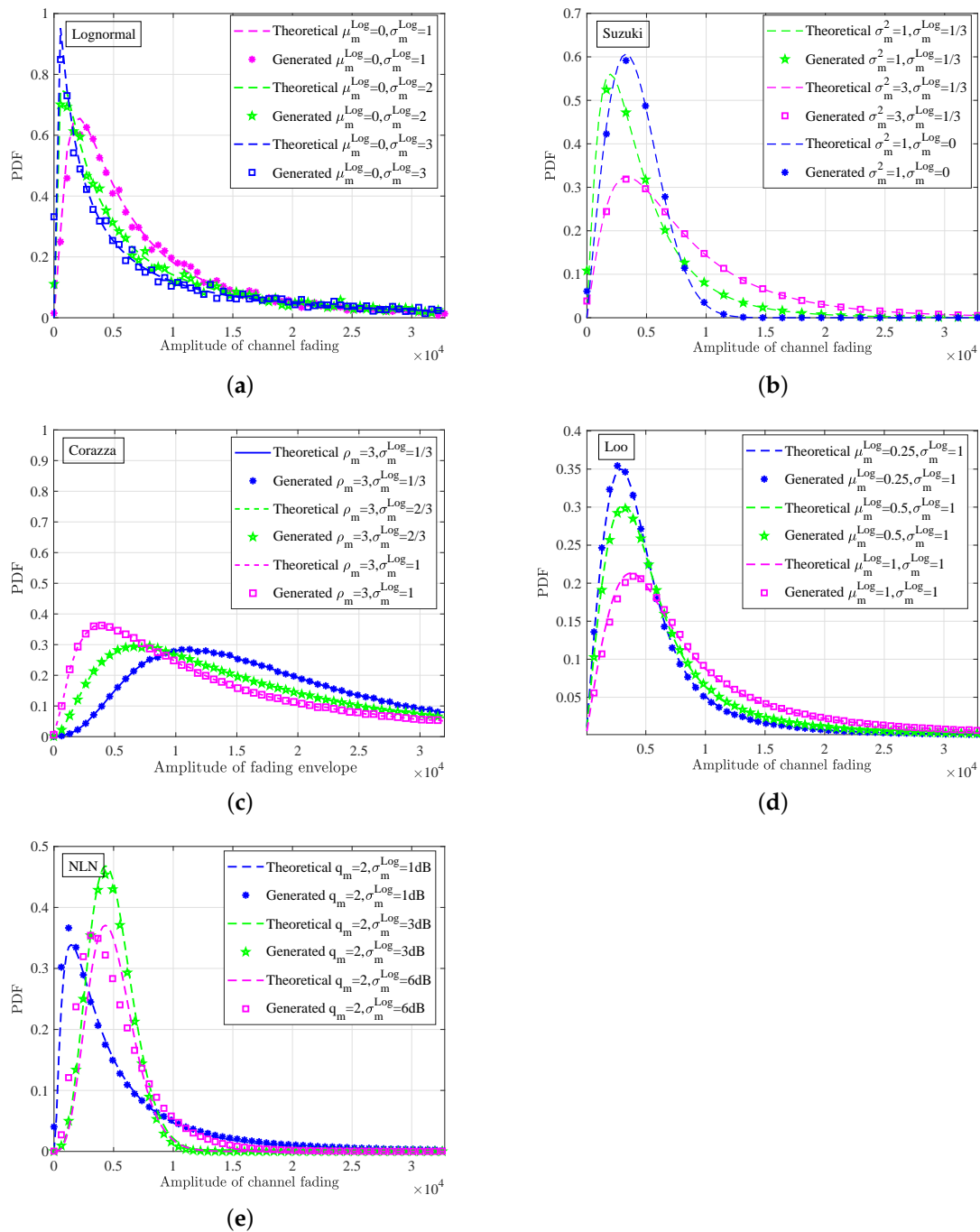


Figure 12. PDFs of (a) lognormal, (b) Suzuki, (c) Corazza, (d) Loo, and (e) NLN fading envelope.

## 5. Conclusions

This paper proposed an efficient generation method for multiple spatial and temporal correlated fading channels, which is also suitable for the non-stationary channel fading under time-variant scattering scenarios. With the TD-based hardware generation architecture, a greedy-CORDIC-based method was developed to perform the trigonometric, exponential, and natural logarithm operations with less RAM resource and better real-time performance than traditional methods. Moreover, the autocorrelation and cross-correlation of the output channel fading can be controlled flexibly by the users. To generate a single-channel fading on a Xilinx FPGA (XC7K325TFFG900) chip, the hardware resource was reduced from 9.14% to 6.21% over the LUT method, and the latency was cut by 62.5% over the classical CORDIC method. The hardware measurement results also demonstrated that the statistical properties of the output channel fading such as the PDF, mean, and variance were consistent with the theoretical ones.

**Author Contributions:** Conceptualization, C.F. and Q.Z.; funding acquisition, C.F.; investigation, B.H. and T.L.; methodology, K.M. and Q.Z.; software, C.F. and Q.Z.; supervision, Q.Z.; validation, S.F. and Z.Z.; writing—original draft, C.F. and Q.Z.; writing—review and editing, Z.Z. and B.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded in part by the National Natural Science Foundation of China under Grant No. 62271250, in part by the Natural Science Foundation of Jiangsu Province, No. BK20211182, in part by the open research fund of the National Mobile Communications Research Laboratory, Southeast University, No. 2022D04, and in part by the Key Technologies R&D Program of Jiangsu (Prospective and Key Technologies for Industry) under Grants BE2022067 and BE2022067-2.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** We thank Thomas Zemen of the Austrian Institute of Technology for his valuable guidance and suggestions during the writing, revision, and improvement of the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hofer, M.; Xu, Z.N.; Vlastaras, D.; Schrenk, B.; Loschenbrand, D.; Tufvesson, F.; Zemen, T. Real-Time Geometry-Based Wireless Channel Emulation. *IEEE Trans. Veh. Technol.* **2019**, *68*, 1631–1645. [[CrossRef](#)]
2. Shitharth, S.; Yonbawi, S.; Manoharan, H.; Alahmari, S.; Yafoz, A.; Mujlid, H. Physical Stint Virtual Representation of Biomedical Signals With Wireless Sensors Using Swarm Intelligence Optimization Algorithm. *IEEE Sens. J.* **2023**, *23*, 3870–3877. [[CrossRef](#)]
3. Mihai, S.; Yaqoob, M.V.; Hung, D.; Davis, W.; Tokakel, P.; Raiza, M. Digital Twins. A Survey on Enabling Technologies, Challenges, Trends and Future Prospects. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 2255–2291. [[CrossRef](#)]
4. Wu, J.; Yang, Y.; Cheng, X.; Zuo, H.; Cheng, Z. The Development of Digital Twin Technology Review. In Proceedings of the 2020 Chinese Automation Congress, Shanghai, China, 6–8 November 2020; pp. 4901–4906.
5. Dakić, A.; Hofer, M.; Rainer, B.; Zelenbaba, S.; Bernadó, L.; Zemen, T. Real-Time Vehicular Wireless System-Level Simulation. *IEEE Access* **2021**, *9*, 23202–23217. [[CrossRef](#)]
6. Zelenbaba, S.; Rainer, B.; Hofer, M.; Zemen, T. Wireless Digital Twin for Assessing the Reliability of Vehicular Communication Links. In Proceedings of the 2022 IEEE Globecom Workshops, Rio de Janeiro, Brazil, 4–8 December 2022; pp. 1034–1039.
7. Yang, Y.; Zhu, Q.; Feng, R.; Fang, C.; Duan, F. High-Efficient Ray-based Hardware Emulator for UAV Channel Digital Twin. In Proceedings of the IEEE 21st International Conference on Communication Technology, Tianjin, China, 13–16 October 2021; pp. 1486–1490.
8. Das, S.; Bhattacharya, A. Application of the Mixture of Lognormal Distribution to Represent the First-Order Statistics of Wireless Channels. *IEEE Syst. J.* **2020**, *14*, 4394–4401. [[CrossRef](#)]
9. Alimohammad, A.; Fard, S.F.; Cockburn, B.F. Hardware implementation of Nakagami and Weibull variate generators. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2012**, *20*, 1276–1284. [[CrossRef](#)]
10. Gutiérrez, C.A.; Gutiérrez-Mena, J.T.; Luna-Rivera, J.M.; Campos-Delgado, D.U.; Velázquez, R.; Pätzold, M. Geometry-Based Statistical Modeling of Non-WSSUS Mobile-to-Mobile Rayleigh Fading Channels. *IEEE Trans. Veh. Technol.* **2018**, *67*, 362–377. [[CrossRef](#)]

11. Wadkar, S.S.; Das, B.P.; Meher, P.K. Low latency scaling-free pipeline CORDIC architecture using augmented Taylor series. In Proceedings of the 2019 IEEE International Symposium on Smart Electronic Systems, Rourkela, India, 16–18 December 2019; pp. 312–315.
12. Silveira Santos Filho, J.C.; Yacoub, M.D. On the simulation and correlation properties of phase-envelope nakagami fading processes. *IEEE Trans. Commun.* **2009**, *57*, 906–909. [[CrossRef](#)]
13. Zhu, B.C. Asymptotic Performance of Composite Lognormal-X Fading Channels. *IEEE Trans. Commun.* **2018**, *66*, 6570–6585. [[CrossRef](#)]
14. Huang, P.D.; Rajan, D.; Camp, J. Weibull and Suzuki fading channel generator design to reduce hardware resources. In Proceedings of the 2013 IEEE Wireless Communications and Networking Conference, Shanghai, China, 7–10 April 2013; pp. 3443–3448.
15. Li, B.; Wang, W.; Wang, Y.; Hu, X. Interference analysis of the integrated satellite-terrestrial networks in the composite fading channel. In Proceedings of the 2017 9th International Conference on Advanced Infocomm Technology, Chengdu, China, 22–24 November 2017; pp. 275–278.
16. Li, Z.; Wang, C.X.; Huang, J. A non-stationary GBSM for 6G LEO satellite communication systems. In Proceedings of the 2021 IEEE/CIC International Conference on Communications in China, Xiamen, China, 28–30 July 2021; pp. 493–498.
17. Liu, T.; Sun, B.W.; Li, Z.G.; Dou, Z. Analysis of channel characteristics and channel model for satellite communication system. In Proceedings of the 2016 IEEE International Conference on Electronic Information and Communication Technology, Harbin, China, 20–22 August 2016; pp. 166–169.
18. Miao, M.K.; Li, X.F. Parameter Estimation of the Lognormal-Rician Channel Model Using Saddlepoint Approximation. *IEEE Access* **2020**, *8*, 152924–152931. [[CrossRef](#)]
19. Rafiq, G.; Pätzold, M. On the statistical properties of the capacity of OSTBC Nakagami-lognormal MIMO channels. In Proceedings of the 2010 4th International Conference on Signal Processing and Communication Systems, Gold Coast, QLD, Australia, 13–15 December 2010; pp. 1–5.
20. Zhu, Q.M.; Li, H.; Fu, Y.; Wang, C.X.; Tian, Y.; Chen, X.M.; Wu, Q.H. A novel 3D non-stationary wireless MIMO channel simulator and hardware emulator. *IEEE Trans. Commun.* **2018**, *66*, 3865–3878. [[CrossRef](#)]
21. Zhao, Z.K.; Zhu, Q.M.; Mao, K.; Liu, W.Q.; Li, N.; Yan, S.Y.; Huang, W. An Efficient Hardware Generator for Massive Non-Stationary Fading Channels. In Proceedings of the 2020 IEEE Globecom Workshops, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
22. Alimohammad, A.; Fard, S.F. A Compact Architecture for Simulation of Spatio-Temporally Correlated MIMO Fading Channels. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2014**, *61*, 1280–1288. [[CrossRef](#)]
23. V'azquez, J.; Vela-Garcia, L.; Guti'érrez, C.A.; Parra-Michel, R. A reconfigurable hardware architecture for the simulation of Rayleigh fading channels under arbitrary scattering conditions. *AEU-Int. J. Electron. Commun.* **2015**, *69*, 1–13. [[CrossRef](#)]
24. Kumar, P.A. FPGA implementation of the trigonometric functions using the CORDIC algorithm. In Proceedings of the 2010 4th International Conference on Signal Processing and Communication Systems, Coimbatore, India, 15–16 June 2019; pp. 894–900.
25. Zhu, B.Z.; Lei, Y.W.; Peng, Y.X.; He, T.T. Low Latency and Low Error Floating Point Sine/Cosine Function Based TCORDIC Algorithm. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2017**, *64*, 892–905. [[CrossRef](#)]
26. Changela, A.; Zaveri, M.; Lakhlani, A. FPGA implementation of asynchronous mousetrap pipelined radix-2 CORDIC algorithm. In Proceedings of the 2018 International Conference on Current Trends towards Converging Technologies, Coimbatore, India, 1–3 March 2018; pp. 252–258.
27. Alimohammad, A.; Fard, S.F.; Cockburn, B.F. Hardware implementation of Rayleigh and Rician variate generators. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2011**, *19*, 1495–1499. [[CrossRef](#)]
28. Beaulieu, N.C.; Xie, J. A Novel Fading Model for Channels With Multiple Dominant Specular Components. *IEEE Wirel. Commun. Lett.* **2015**, *4*, 54–57. [[CrossRef](#)]
29. Entezari, A.; Tadaion, A. Performance Analysis of Cellular Networks Under Nakagami-Lognormal Composite Fading Channel Utilizing Interference Management Technique. In Proceedings of the Electrical Engineering, Mashhad, Iran, 8–10 May 2018; pp. 679–683.
30. Kumari, D.L.; Giri Prasad, M.N.; Shakeer, S.M. Performance of MIMO systems using Maximal Ratio Combining over Weibull Fading Channel. In Proceedings of the 2018 International Conference on Recent Innovations in Electrical, Electronics and Communication Engineering, Bhubaneswar, India, 27–28 July 2018; pp. 1981–1986.
31. Meher, P.K.; Valls, J.; Juang, T.; Sridharan, K.; Maharatna, K. 50 Years of CORDIC: Algorithms, Architectures, and Applications. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2009**, *56*, 1893–1907. [[CrossRef](#)]
32. Zhu, Q.M.; Huang, W.; Mao, K.; Zhong, W.Z.; Hua, B.Y.; Chen, X.M.; Zhao, Z.K. A Flexible FPGA-Based Channel Emulator for Non-Stationary MIMO Fading Channels. *Appl. Sci.* **2020**, *10*, 4161. [[CrossRef](#)]
33. Zhu, Q.; Mao, K.; Song, M.; Chen, X.; Hua, B.; Zhong, W. Map-based channel modeling and generation for U2V mmwave communication. *IEEE Trans. Veh. Technol.* **2022**, *71*, 8004–8015. [[CrossRef](#)]
34. Mao, K.; Zhu, Q.; Song, M.; Li, H.; Ning, B.; Pedersen, G.F. Machine learning-based 3D channel modeling for U2V mmwave communications. *IEEE Internet Things J.* **2022**, *9*, 17592–17607. [[CrossRef](#)]
35. Zhu, Q.M.; Zhao, Z.K.; Mao, K.; Chen, X.M.; Liu, W.Q.; Wu, Q.H. A Real-Time Hardware Emulator for 3D Non-Stationary U2V Channels. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 3951–3964. [[CrossRef](#)]



36. Zhou, S.; Zhu, Q.; Dai, X.; Liu, X.; Chen, X. Analysis and simulation for temporal correlated composite fading channels. In Proceedings of the 2014 3rd Asia-Pacific Conference on Antennas and Propagation, Harbin, China, 26–29 July 2014; pp. 690–693.
37. Zhu, Q.; Zhao, Y.; Huang, Y.; Lin, Z.; Wang, L.H.; Bai, Y. Demo abstract: An UAV-based 3D spectrum real-time mapping system. In Proceedings of 2022 IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops, New York, NY, USA, 2–5 May 2022; pp. 1–2.
38. Li, H.; Chen, X.; Mao, K.; Zhu, Q.; Qiu, Y.; Ye, X. Air-to-ground path loss prediction using ray tracing and measurement data jointly driven DNN. *Comput. Commun.* **2022**, *196*, 268–276. [[CrossRef](#)]
39. Zhu, Q.M.; Lv, W.H.; Xu, D.Z.; Chen, X.M.; Xu, Y.H. A novel simulator for correlated generalized-K composite fading channels. In Proceedings of the ISAPE2012, Xi'an, China, 22–26 October 2012; pp. 475–477.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.